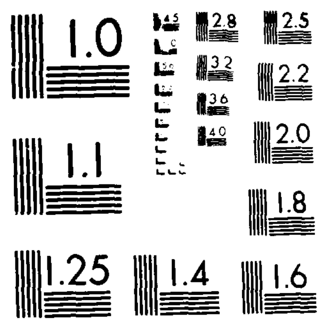
AD-A199 591 CONDITIONS ON SIMULATION PARAMETERS IN SURFACE SCATTERING (CONTINUATION)(U) TEXAS UNIV AT ARLINGTON A K FUNG ET AL. JAN 88 N00014-87-K-0751 1/1

UNCLASSIFIED F/G 20/14 NL

END
DATE
FILMED
11 88



MICROCOPY RE OLUT

(2)

Conditions on *Simulation Parameters in Surface Scattering*

(Continuation)

Quarterly Report

From November 1987 to January 1988

Prepared for DARPA under contract N00014-87-K-0751

by

A.K.Fung and P.M. Chen

Wave Scattering Research Center

Box 19016

University of Texas at Arlington

Arlington, TX 76019

88-1360

TABLE OF CONTENTS

ABSTRACT

This report is a continuation of the previous study on simulation parameters. Additional cases studied in this report indicate that the minimum number of points per wavelength or per correlation length whichever is the smaller is around 5, the minimum number of surface samples is 25 and the minimum width of the illuminated area is 7 λ or 7L whichever is the larger. These studies are still incomplete in that we have not considered enough variations in the surface rms height, surface rms slope and the ratio of the incident wavelength to the surface correlation.

1.0 INTRODUCTION

This is the second quarterly report on the Contract N00014-87-K-0751. In our first report [Fung et al, 1987] we have examined the problem of surface generation, the calculation of the backscattering coefficient and the conditions on the selection of simulation parameters for certain types of surface. This quarter we have an update on our surface simulation program and we continue our studies on the selection of simulation parameters for additional surface types. This means that finer divisions of cases are considered to realize greater efficiency in the simulation program. The need for an efficient program was pointed out in our previous quarterly report.

In Appendix A we provide an updated version of our surface generation program where a more efficient fast Fourier transform algorithm is used for filter weight calculations and surface generation.

Based upon our studies in the previous quarter we have found that we need a minimum of 4-5 points per wavelength λ ; with a Gaussian antenna pattern $\exp(-x^2/g^2)$ we need an effective illuminated area 2g equals to about 3 correlation lengths L and the edge of the illuminated area should be taken to be the point where the antenna pattern has decayed to 10^{-3}. In all cases 50 surface samples were used to obtain a statistically averaged value. In these studies surface parameters with rms surface height σ of 0.5 unit and $\lambda = L = 2$ units were used. In this quarter we continue the same study by examining the effects of surface parameters on the selection of D, the width of the illuminated area, and the number of points per wavelength, λ, or correlation length, L, whichever is the smaller and the total number of samples, N, under different

combinations of the following surface conditions: (1) $L > \lambda$, (2) $L < \lambda$, (3) normalized rms height, $k\sigma = 2\pi\, \sigma / \lambda < 1$,(4) $k\sigma > 1$. The same antenna pattern is used in these studies as in the previous quarter.

2.0 SELECTION OF SIMULATION PARAMETERS

In what follows N the number of surface samples, D the width of the illuminated area, and the number of points per correlation length L or per wavelength λ whichever is the smaller are considered under different surface conditions. It is anticipated that D is a function of λ and L but it is not immediately obvious whether it should depend only on λ or L whichever is the larger. N is expected to be a function of the normalized rms height $k\sigma$. The number of points per wavelength or correlation length whichever is the smaller may depend on the relative size of λ and L. It is possible that the required number of points per correlation length may be different from the number of points per wavelength .

2.1 Number of Surface Points Per Wavelength

To investigate the number of points per wavelength we must make sure that the requirements on other surface parameters are satisfied. In Figure 1 we have chosen L greater than λ , D greater than 10 L and an N equal to 50. We shall show in subsequent sections that both D and N are sufficiently large. In addition, since L is greater than λ, it automatically contains enough number of points if λ contains enough number of points. The number of points per wavelength considered ranges from 1.3 to 8 in Figure 1. From Figure 1 (a) it is seen that the backscattering coefficient curves begin to converge when the number per wavelength exceeds 2.6. The difference between 4 points and 8 points per wavelength is around 2 dB or less. The same observation can be made when the rms surface height is increased from 0.3 to 0.6 in Figure 1(b). Further increase in the rms surface height σ to 1.2 in Figure 1(c) does not seem to cause any change in the number of points required per wavelength.

Thus, we conclude that depending upon the amount of error one can tolerate the required number of points per wavelength should be between 4 and 8.

2.2 Number of Surface Points Per Correlation Length

To investigate the number of surface points per correlation length we choose the wavelength λ greater than the correlation length L. In Figure 2(a) the number of surface points per correlation length is varied from 0.8 to 8. It is seen that convergence in the backscattering curves occurs when there are more than two points per correlation length. It appears that 4 to 8 points per correlation length seem to produce satisfactory results. Then in Figure 2(b) the rms surface height is increased from 0.15 to 0.3 unit. Similar observations remain applicable. Next, we keep other parameters the same and increase the correlation length from 1 to 2 units and maintain the same surface slope by doubling the surface rms height in Figure 2(c) and 2(d). In these cases comparable accuracy to previous cases can be obtained by requiring more than 5 points per correlation length. It is not clear whether this somewhat more stringent requirement comes from a larger L or a larger rms height or the higher order surface frequency components resulting from the use of finite record length in surface generation.

2.3 Width of the Illuminated Area

Two different cases are considered when study the width of the illuminated area : (1) when $L > \lambda$ and (2) when $\lambda > L$. In case (1) we want to examine how many L should be contained in the illuminated area D so that the scattering coefficient calculated agrees with that from an infinitely large surface. Similarly, in case (2) we want to find out how many λ should be

contained in D to achieve the scattering coefficient from an infinite area.

Figure 3 shows the case with $L > \lambda$. All surface parameters are given in the figure where K is the wave number and 'sig' stands for σ . The width of the illuminated area is varied from D = 5L to 8L. It is seen that the backscattering curves appear to converge when D is greater than 7L. In all the computations 8 or more points per wavelength are used and a minimum of 45 scattering coefficient samples has been averaged. The minimum required size of the illuminated area seems to be 7L depending upon the amount of error one can tolerate. In addition, the theoretical scattering coefficient predicted by the Kirchhoff surface scattering model [Ulaby et al, 1982] is also plotted in Figure 3 to provide a reference. The polarization used in this report is HH polarization. Since the Kirchhoff scattering model has no polarization dependence, in general we expect fair agreement for small angles of incidence up to about 25 degrees and higher theoretical than simulated values at large angles of incidence [Fung and Pan, 1987].

Figure 4 shows the case with $\lambda > L$. All surface parameters are given in the figure. Since $\lambda > L$ we seek the width of the illuminated area D in terms of λ and vary D from D=5 λ to 9 λ. When D is greater than or equal to 7 λ , the computed scattering coefficients are practically the same indicating that convergence of results has occurred. Thus, the minimum width of the illuminated area required in this case is 7 λ . Note again that we have used N = 45 which is larger than the minimum required values for the surface samples (to be demonstrated in the nest section) and a minimum of 8 points per correlation length in all the calculations.

2.4 Number of Surface Samples Averaged

In Figure 5 we illustrate the requirement on the number of surface samples needed to achieve an acceptable averaged value for the backscattering coefficient. Because the selected case is a slightly rough surface with a small normalized rms surface height, $k\sigma$ of 0.24, there is a significant contribution from the coherent component of the backscattering coefficient. This explains the peaking near the nadir region. For the case shown convergence appears to occur after the number of samples N reaches 25. Here again to be sure that other surface and system parameters do not affect the results obtained we use values which exceed the minimum requirements for all of them. These values are given in the figure except the number of points per correlation length which is 8.

In Figure 5 we have plotted the total scattering coefficient which is the sum of the coherent plus the incoherent scattering coefficients. When $k\sigma$ is larger than unity the coherent contribution becomes negligible. For this case and the case of $L > \lambda$ we illustrate the requirement on N in Figure 6. Here N is varied from 5 to 30. It is seen that after N exceeds 20 a gradual convergence is obvious. Depending upon the amount of error one can accept N = 25 seems to be the minimum number of samples needed.

3.0 DISCUSSIONS

From the above studies we note that the minimum number of points per wavelength or per correlation length whichever is the smaller is around 5, the minimum number of surface samples is 25 and the minimum width of the illuminated area is 7 λ or 7L whichever is the larger. The minimum number of points per λ or L may have a dependence on the ratio of λ/L. This has not been studied extensively. In the study of the width of the illuminated area we also have not considered enough variations in surface roughness that is a change in $k\sigma$ and σ/L. In addition, there may be a dependence on how perfect is the generated surface meeting its statistical description. For example, the difference between the required number of points per correlation length in Figure 2(b) and 2(d) may be caused by possible presence of high frequency components in the surface not reflected by the correlation length parameter. To better understand this problem some smoothing operation should be performed on the surface to ensure that there are no unwanted high frequency surface components.

For a statistically described surface there is always a finite probability for it to take on values substantially different from its mean even when the surface statistics perfectly match its required specification.. Furthermore, since we can only deal with **finite** samples, the averaged result may be affected substantially by one or two large samples with very low but finite probability of occurrence. Thus, special care must be exercised in estimating the mean scattering coefficient especially at large angles of incidence.

4.0 REFERENCES

A.K.Fung,G. Gan and P.M. Chen, Conditions on simulation parameters in surface scattering , Quarterly report, DARPA Contract N00014-87-K-0751, Noember, 1987

A. K. Fung and G. W. Pan, A scattering model for perfectly conducting random surfaces, Int. J. Remote Sensing, vol. 8, no.,11, pp. 1579-1593, 1987

F. T. Ulaby, R. K. Moore, and A. K. Fung, Microwave Remote Sensing Active and Passive, Vol. II, Radar Remote Sensing ~~and~~ Surface Scattering and Emission Theory. Reading, MA: Addison-Wesley, 1982.

5.0 FIGURE LEGENDS

1. Dependence of the backscattering coefficient on the number of points per wavelength when N is 50 , the rms surface height is (a) 0.3 unit (b) 0.6 unit and (c) 1.2 units.

2. Dependence of the backscattering coefficient on the number of points per correlation length L with N = 50 and when (a) the rms surface height is 0.15 unit and the correlation length is 1 unit (b) the rms surface height is 0.3 unit and the correlation length is 1 unit (c) the rms surface height is 0.3 unit and the correlation length is 2 units and (d) the rms surface height is 0.6 unit and the correlation length is 2 units.

3. Dependence of the backscattering coefficient on the width of the illuminated area D when the correlation length is larger than the electromagnetic wavelength.

4. Dependence of the backscattering coefficient on the width of the illuminated area D when the correlation length is smaller than the electromagnetic wavelength.

5. Dependence of the backscattering coefficient on the number of surface samples N when the correlation length is smaller than the electromagnetic wavelength.

6. Dependence of the backscattering coefficient on the number of surface samples N when the correlation length is larger than the electromagnetic wavelength.

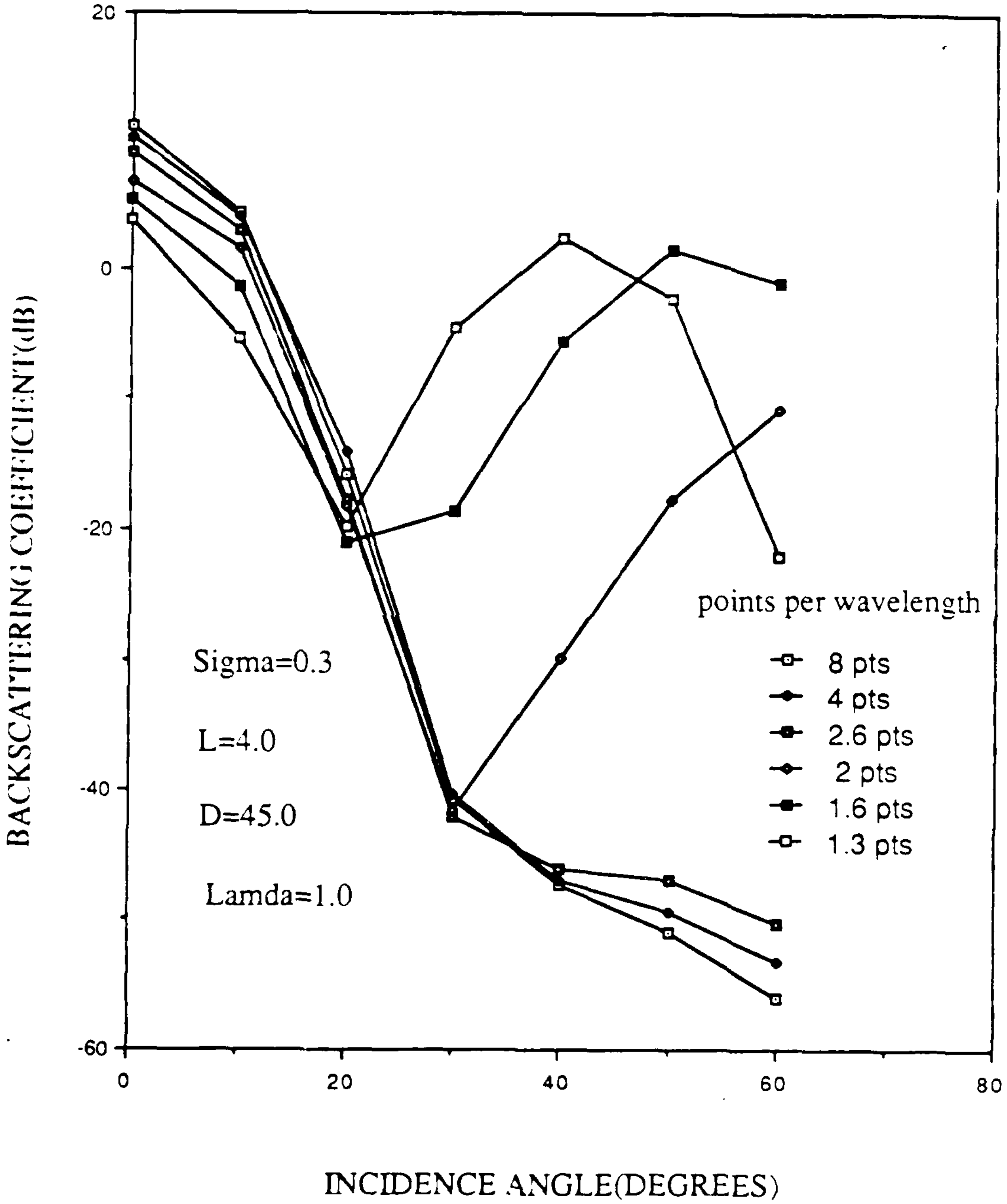


Figure 1(a) Dependence of the backscattering coefficient on the number of points per wavelength when N is 50, the rms surface height is 0.3 unit

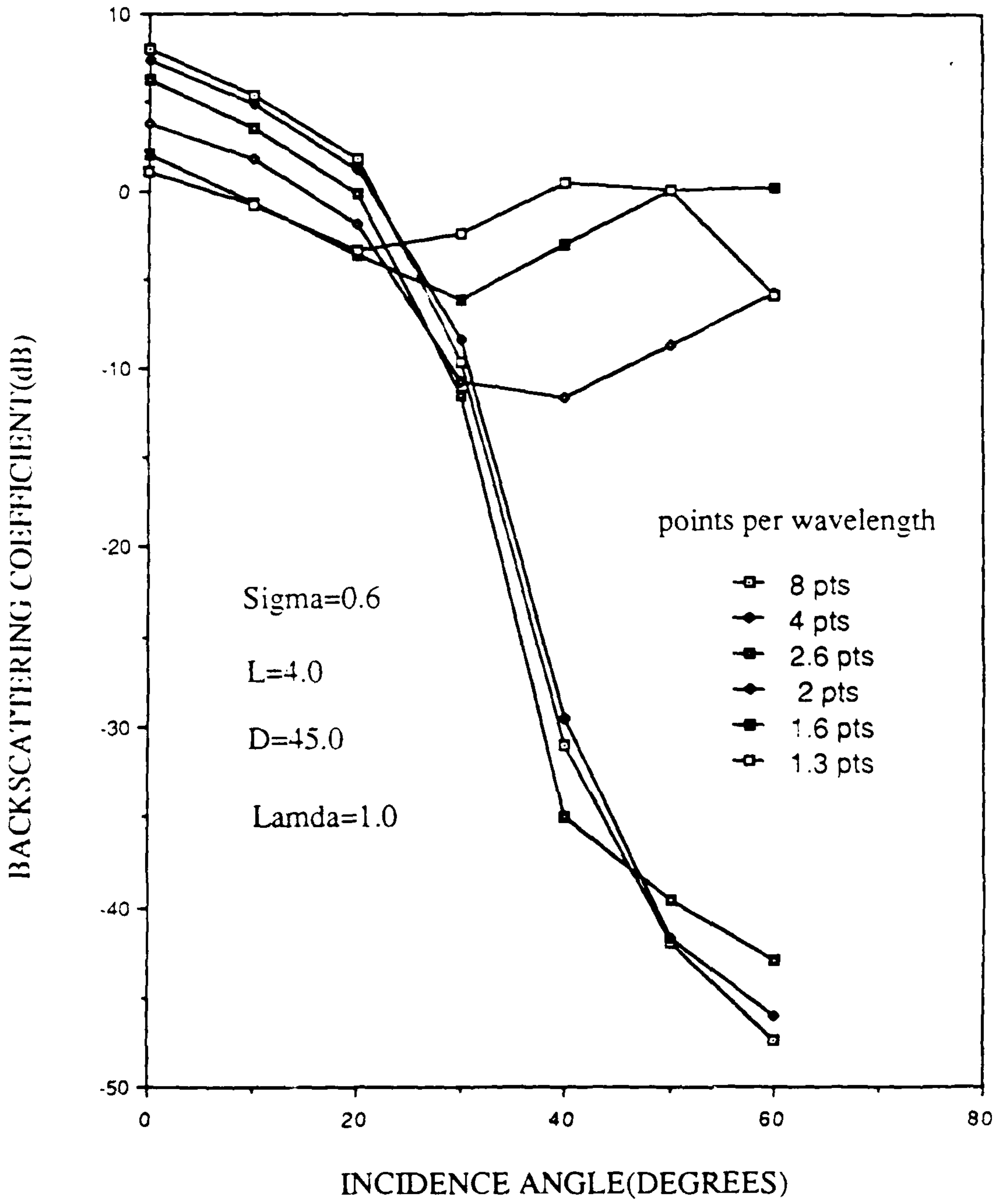


Figure 1(b) Dependence of the backscattering coefficient on the number of points per wavelength when N is 50, the rms surface height is 0.6 unit

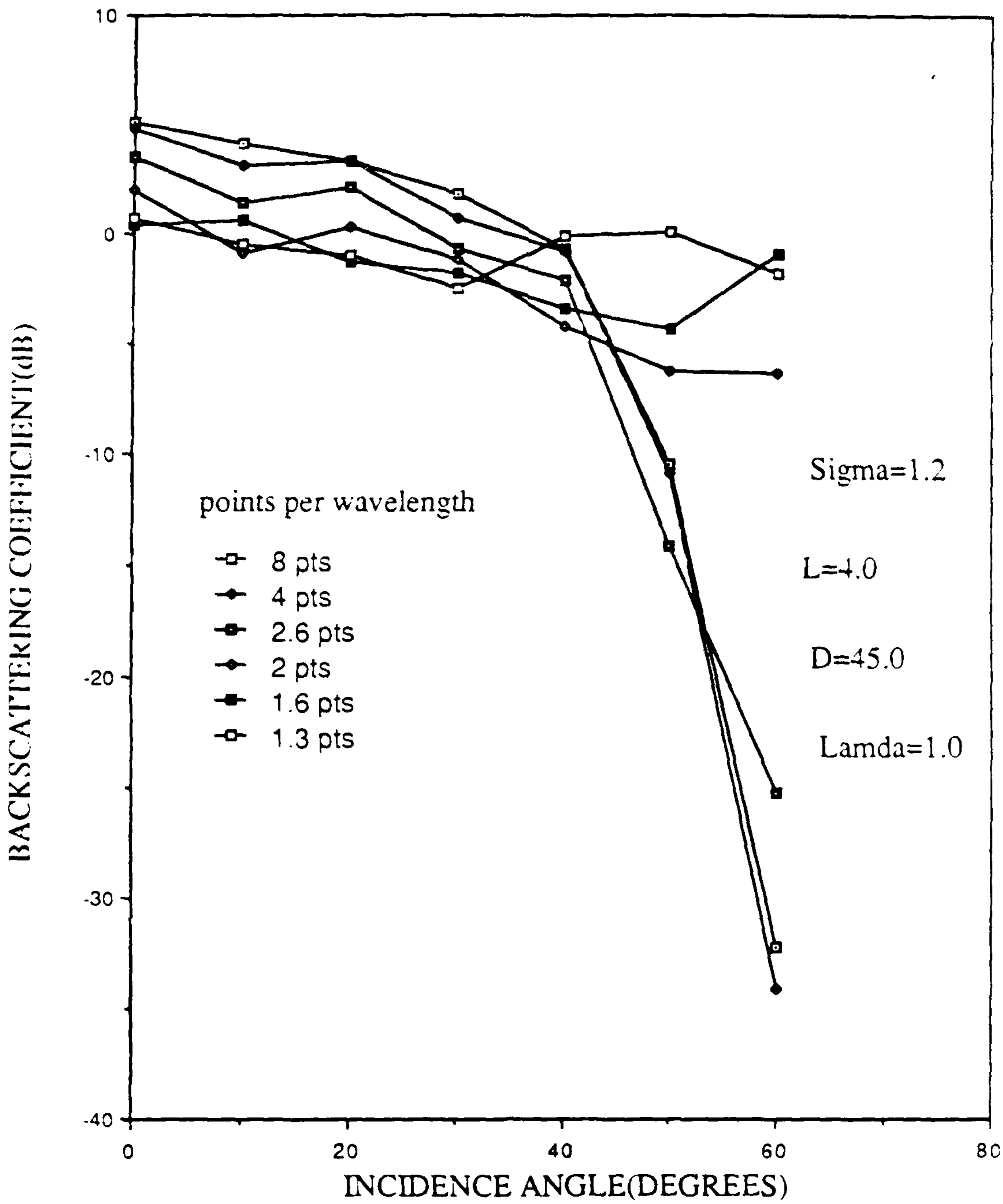


Figure 1(c) Dependence of the backscattering coefficient on the number of points per wavelength when N is 50, the rums surface height is 1.2 units

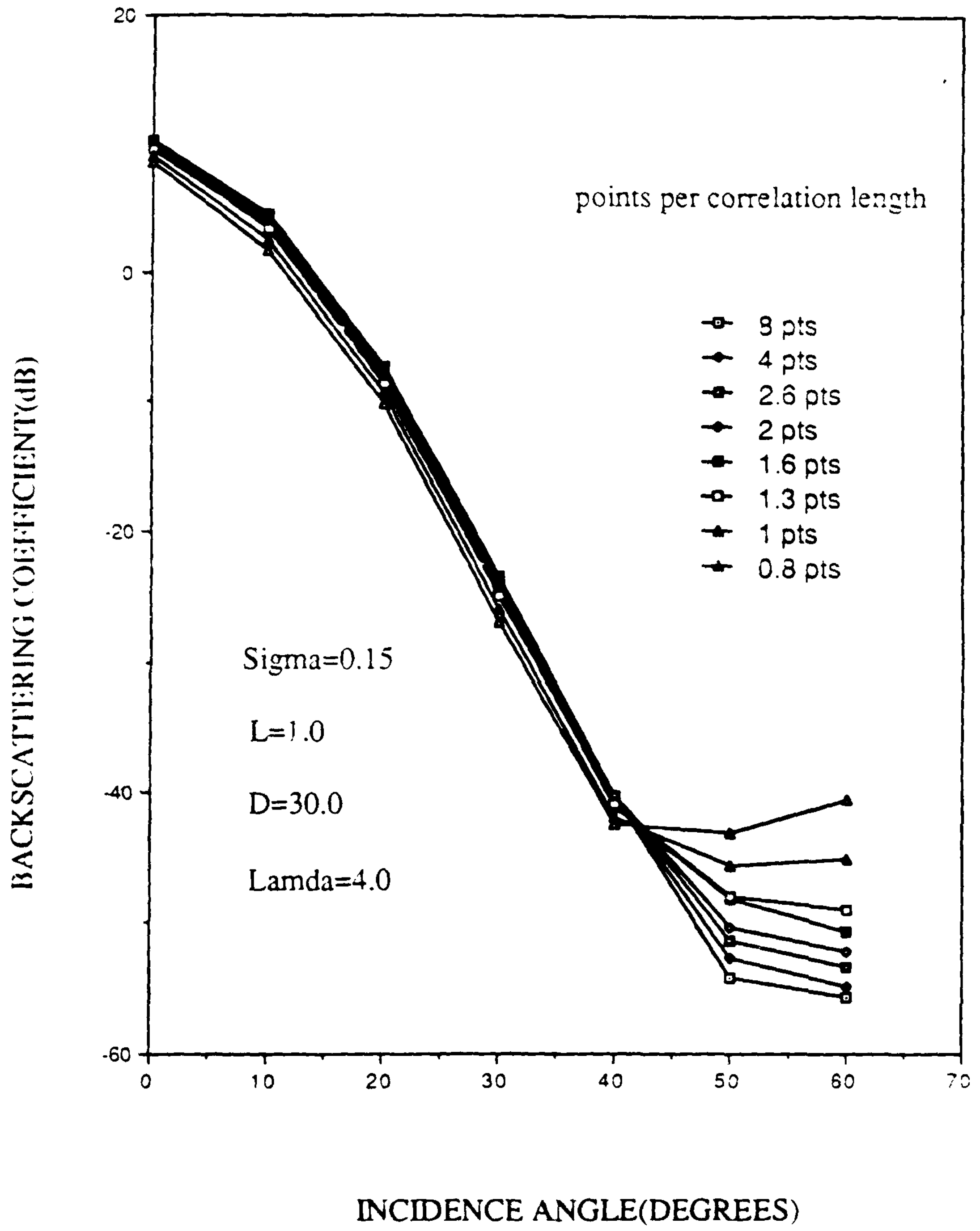


Figure 2(a) Dependence of the backscattering coefficient on the number of points per correlation length L with L = 30 and when the rms surface height is 0.15 unit and the correlation length is 1 unit

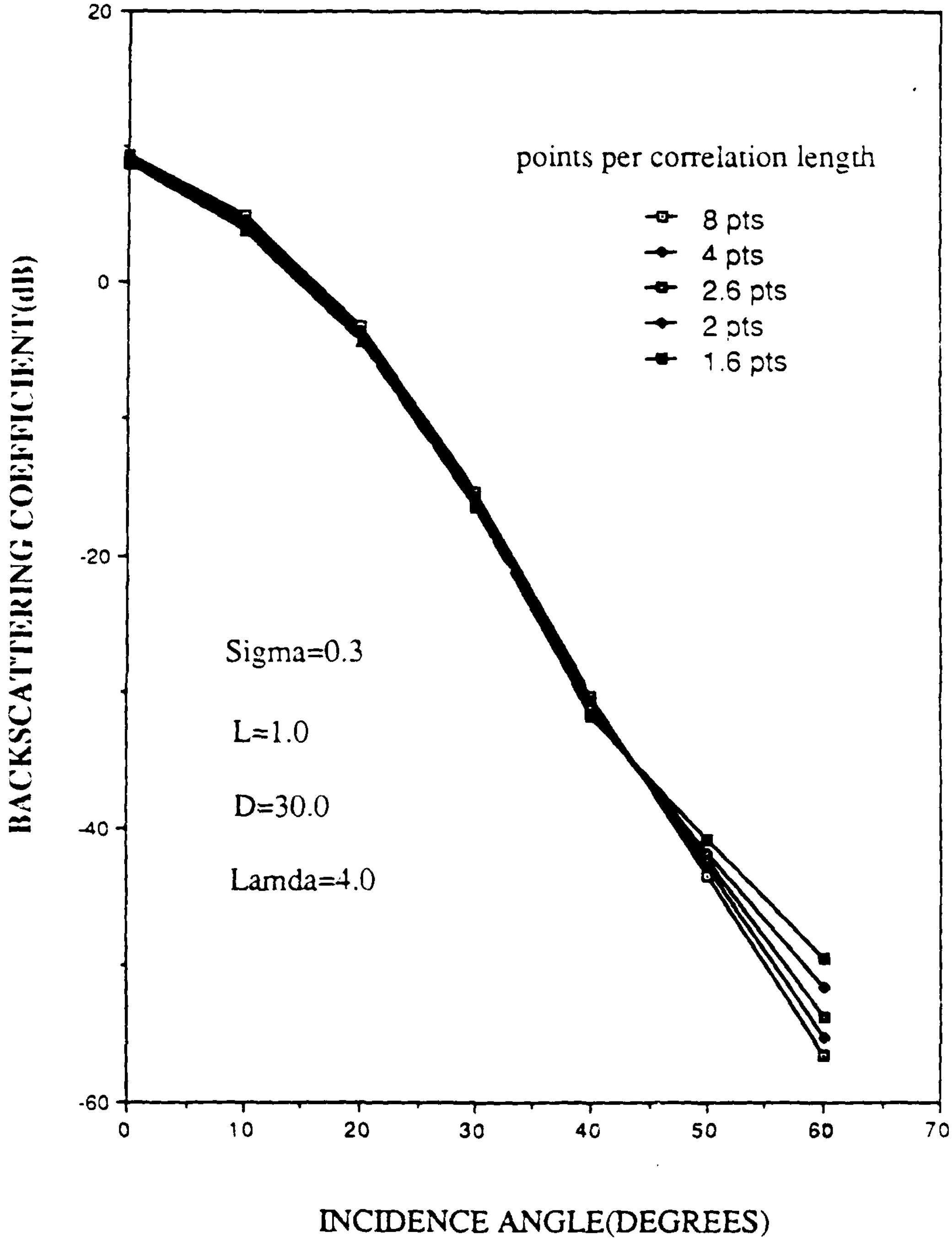


Figure 2(b) Dependence of the backscattering coefficient on the number of points per correlation length L with N = 50 and when the rms surface height is 0.3 unit and the correlation length is 1 unit

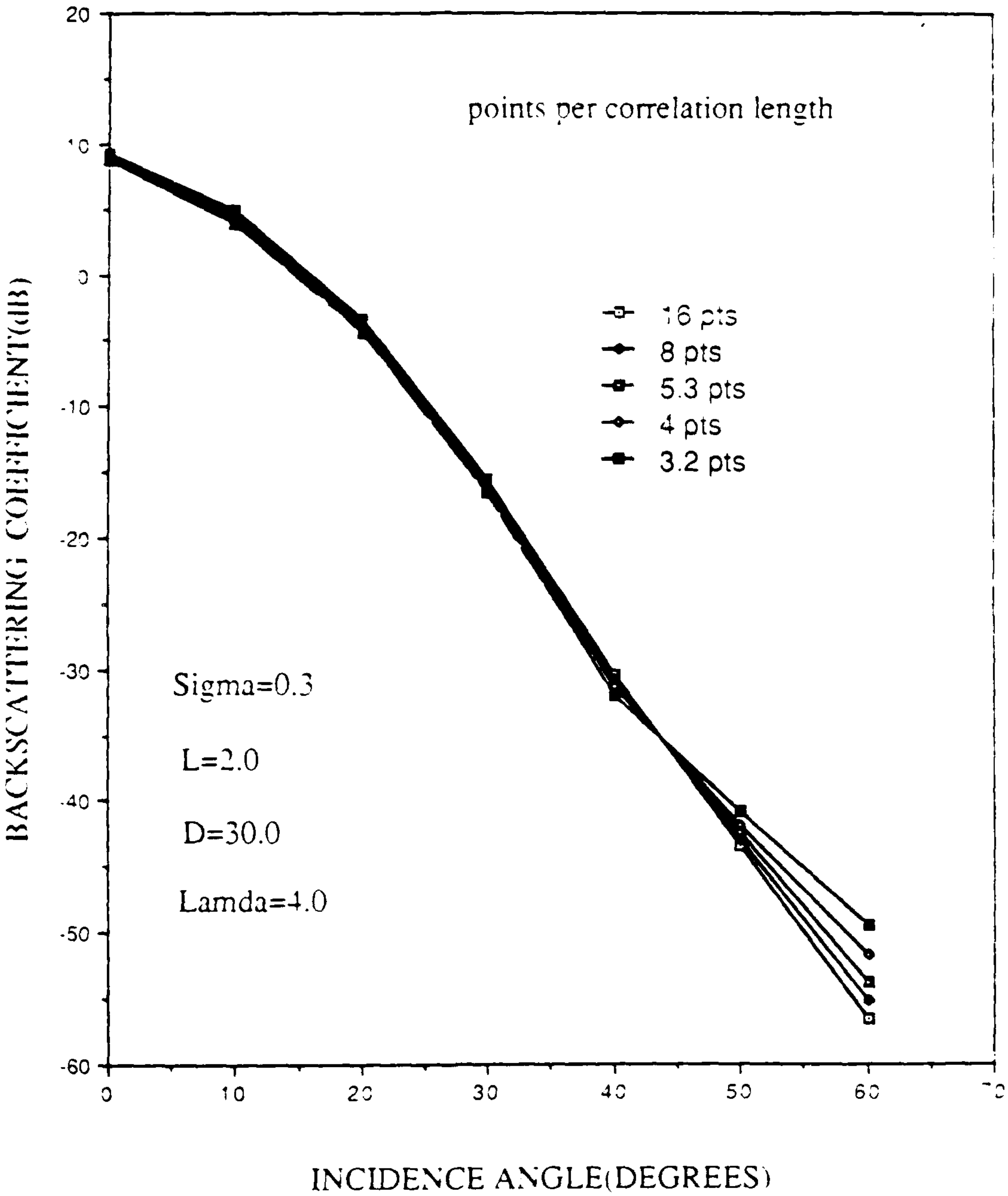


Figure 2(c) Dependence of the backscattering coefficient on the number of points per correlation length L with N = 30 and when the rums surface height is 0.3 unit and the correlation length is 2 units

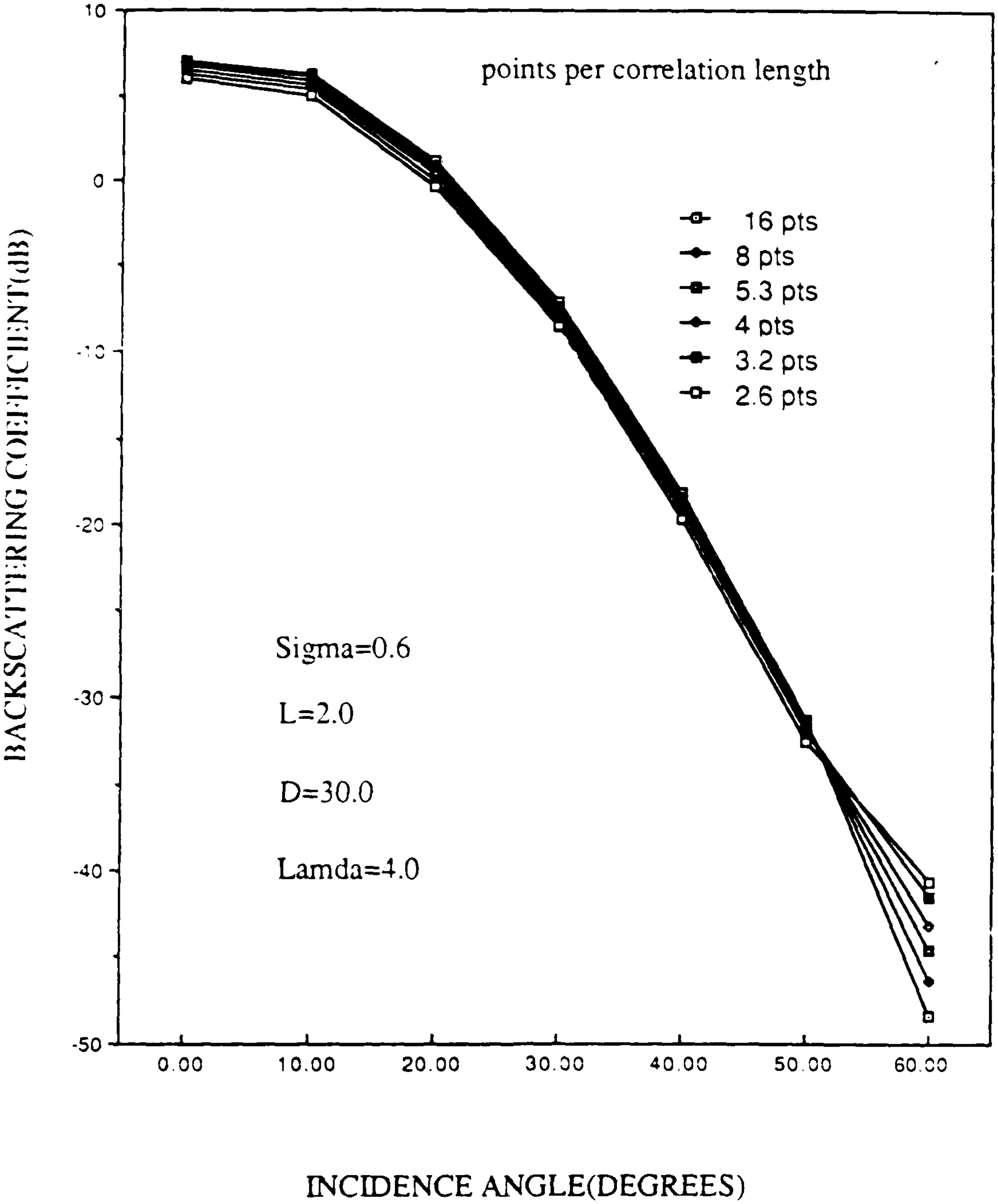


Figure 2(d) Dependence of the backscattering coefficient on the number of points per correlation length L with N = 50 and when the rums surface height is 0.6 unit and the correlation length is 2 units

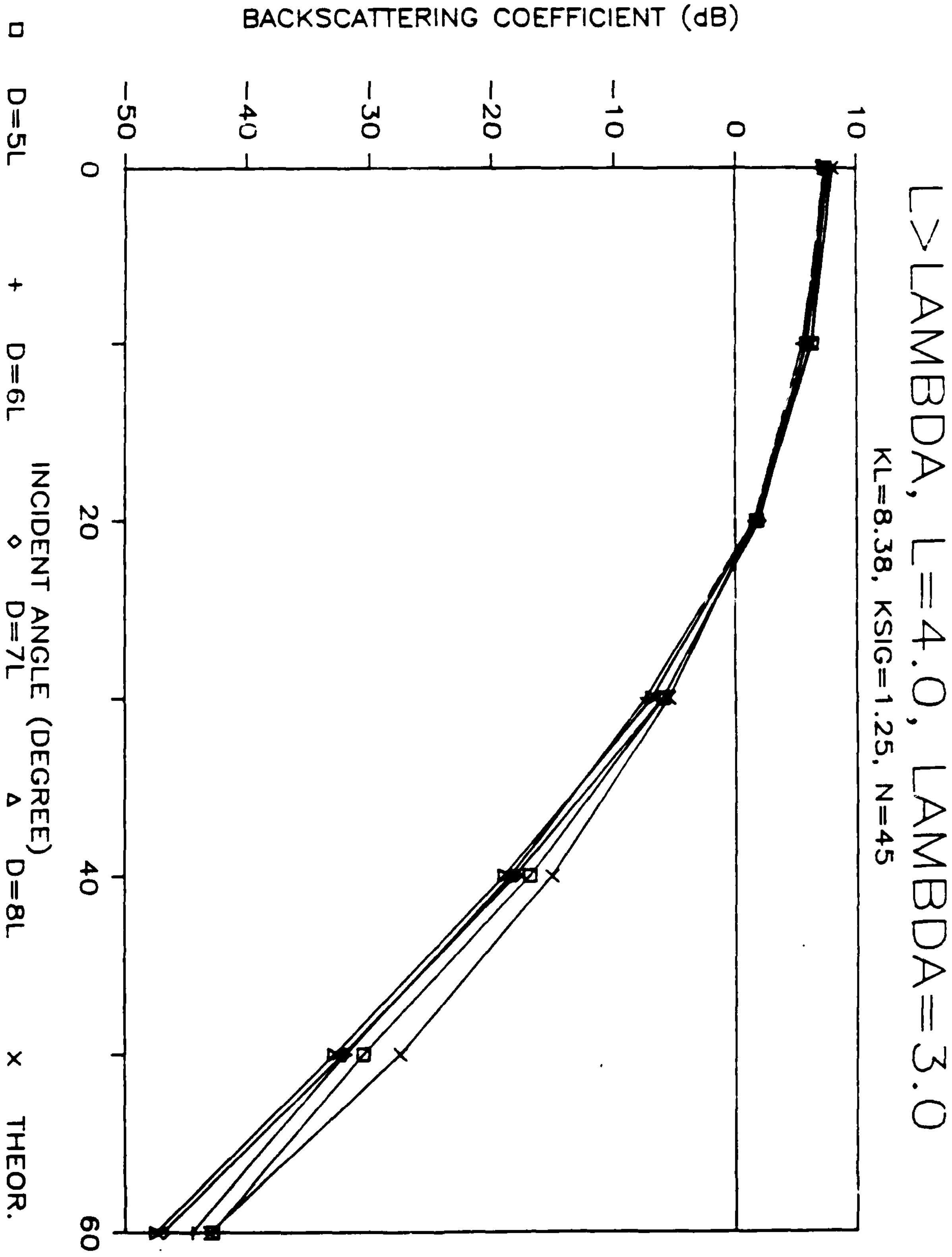


Figure 3 Dependence of the backscattering coefficient on the width of the illuminated area D when the correlation length is larger than the electromagnetic wavelength

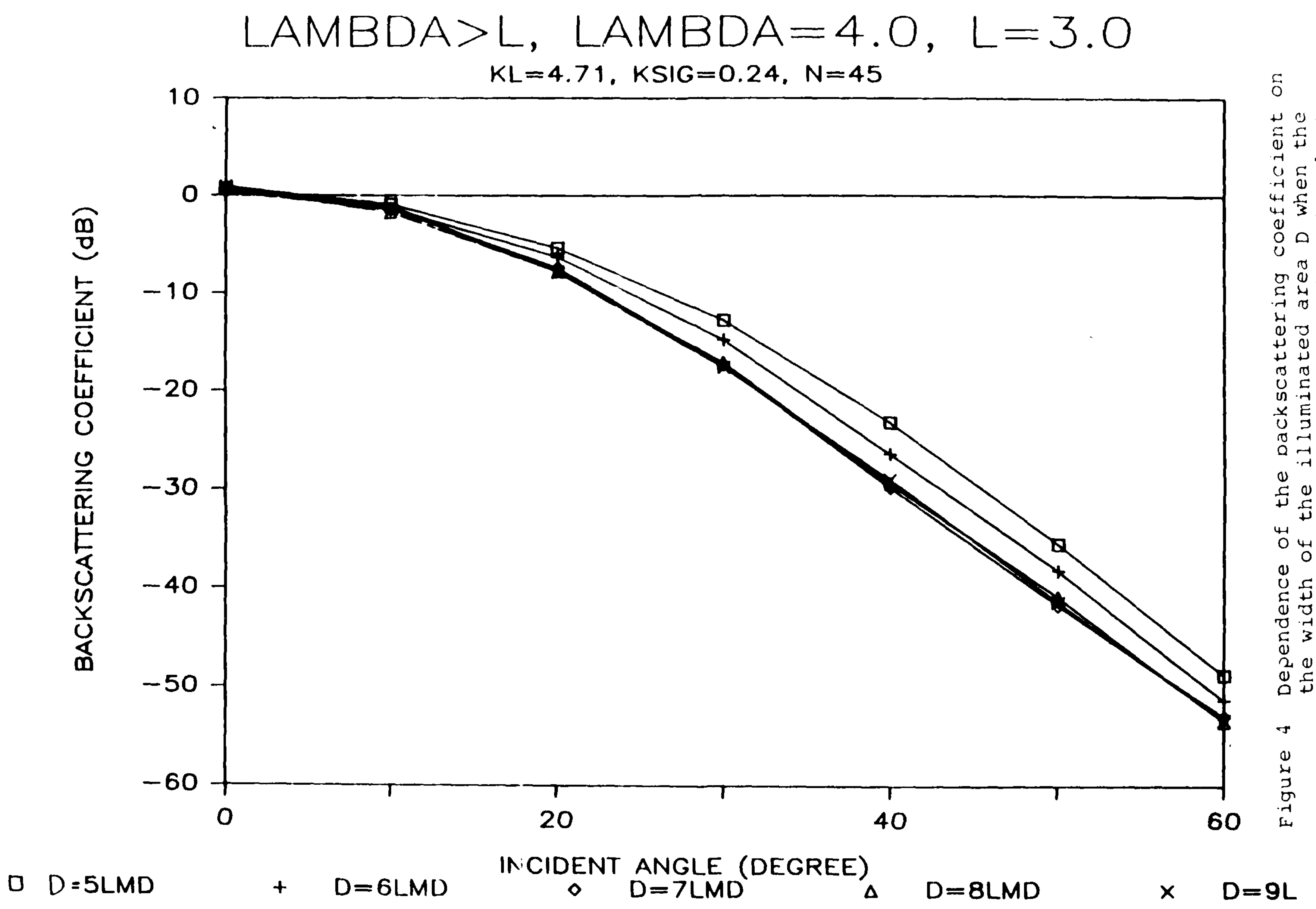


Figure 4 Dependence of the backscattering coefficient on the width of the illuminated area D when the correlation length is smaller than the electromagnetic wavelength

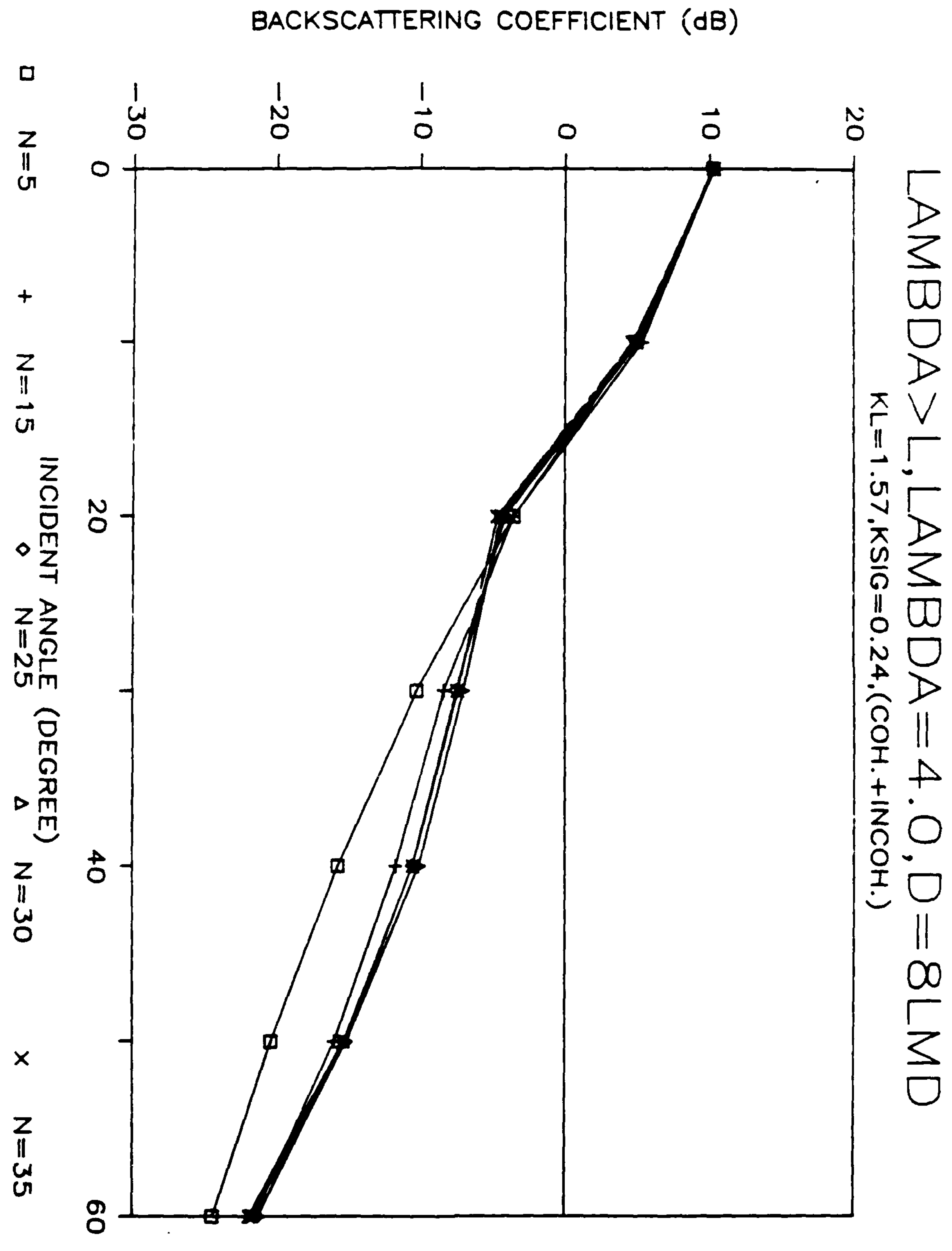


Figure 5 Dependence of the backscattering coefficient on the number of surface samples N when the correlation length is smaller than the electromagnetic wavelength

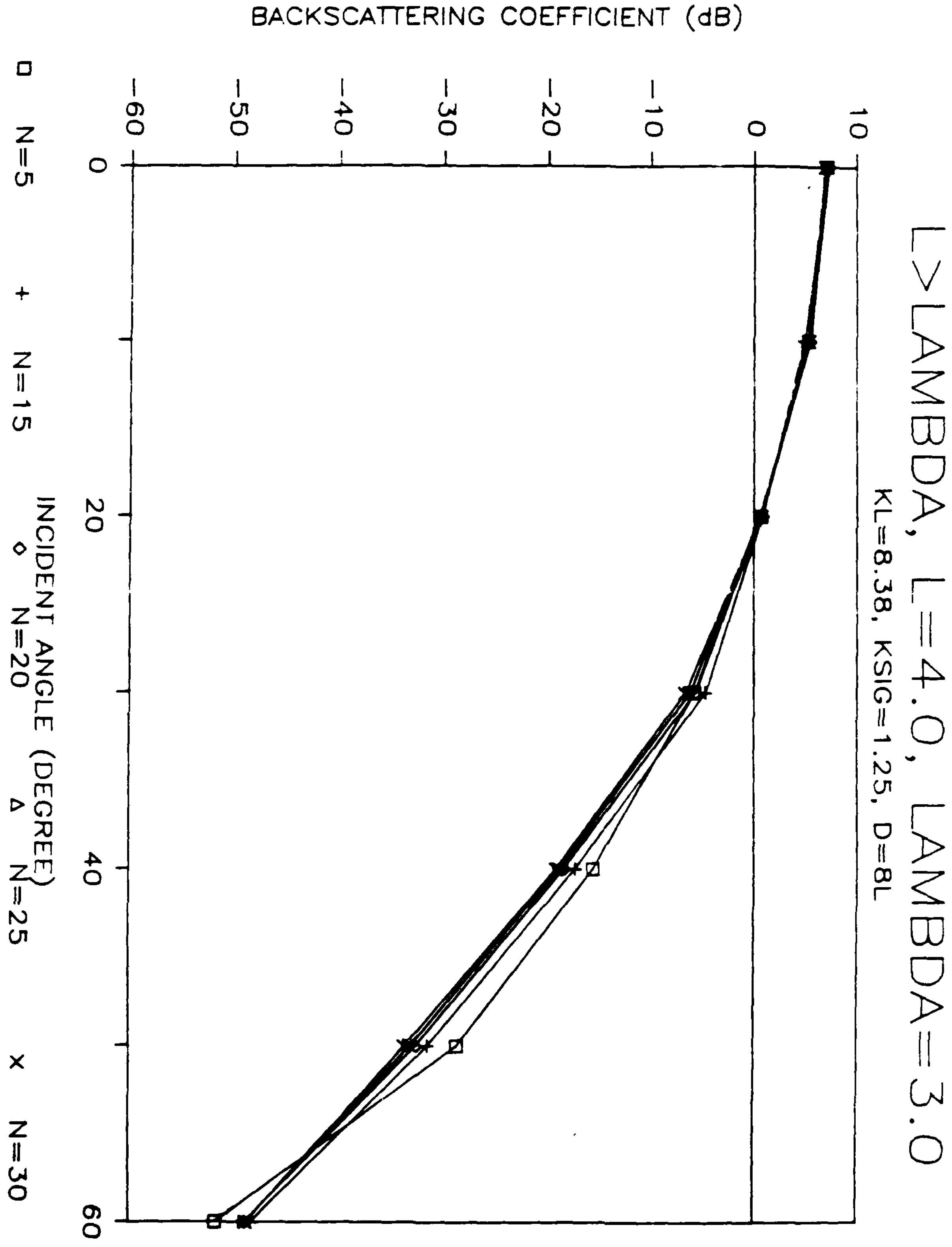


Figure 6 Dependence of the backscattering coefficient on the number of surface samples N when the correlation length is larger than the electromagnetic wavelength

```
C------------------------------------------------------------------C
C    PROGRAM NAME GENSRF.FOR                          C
C    PURPOSE  GENERATION OF RANDOM SURFACE
C
C    PARAMETERS
C    NX  PATCH SIZE OF THE SURFACE IN X-DIRECTION
C    NY  PATCH SIZE OF THE SURFACE IN Y-DIRECTION
C    NWX  SIZE OF THE WEIGHTING FUNCTION IN X-DIR
C    NWY  SIZE OF THE WEIGHTING FUNCTION IN Y-DIR
C       NOTE NWX,NWY MUST BE AN ODD NUMBER
C    NWXH   NWXH=(NWX+1)/2
C    NWYH   NWYH=(NWY+1)/2
C    N,M  THE SIZE CHOSEN FOR THE FFT TO FIND THE WEIGHTING
C       FUNCTION  MUST BE POWER OF 2
C    MX   ACTUAL SIZE NEEDED TO DO FAST COSINE TRANSFORM
C       MX=M/2 + 1
C    NZY  SIZE OF THE RANDOM NUMBER MATRIX BEFORE SMOOTHED
C       BY THE WEIGHTING FUNCTION (Y-DIRECTION)
C    NZX  SIZE OF THE RANDOM NUMBER MATRIX IN THE X-DIRECTION
C       BEFORE SMOOTHING  NOTE THAT THE SETUP HERE IS TO
C       GENERATE A STRIP OF RANDOM SURFACE ALONG X-DIRECTION
C       THUS FOR NN PATCHES, NZX=NN*NX+NWX-1  NZY=NY+NWY-1
C
C------------------------------------------------------------------C
      PARAMETER(NX=300,NY=300,NWXH=33,NWYH=33,NWX=65,NWY=65,NN=1
      PARAMETER(N=512,M=512,MX=257)
      PARAMETER (NZX=NN*NX+NWX-1,NZY=NY+NWY-1)
      REAL S(MX,MX)
      REAL X(M),Y(M)
      REAL WD(N)
      DIMENSION Z(NZY,NZX)
      DIMENSION WA(NWY,NWX)
C
      WD=1
      L=NX
C--------------------------------------------------------------------
C    ASSIGNING CORRELATION LENGTH OF SAMPLE GAUSSIAN CORRELATION
C    FUNCTION  XL=CORR. LENGTH IN X-DIRECTION  YL=CORR. LENGTH
C    IN Y-DIRECTION  NOTE THAT THE LENGTH HERE IS THE NUMBER OF
C    INTERVALS
C--------------------------------------------------------------------
      XL=16.0
      YL=16.0
      NP1=N+1
      NP2=N+2
      NO2=N/2
      NO2P1=NO2+1
      TP=N/2+2
```

```
      NP1=N+1
      NP2=N+2
C-----------------------------------------------------------------------
C     THE FOLLOWING NESTED DO LOOP ASSIGNS THE DIGITIZED CORRELATION
C     VALUES IN THE ARRAY S(MX,MX). NOTE THAT THE CORRELATION IS
C     ALWAYS SYMMETRIC WITH RESPECT TO THE ORIGIN IF STATIONARITY IS
C     IS ASSUMED FOR THE SURFACE. THEREFORE ONLY A QUARTER OF THE TWO-
C     DIMENSIONAL CORRELATION FUNCTIONS IS USED FOR CALCULATION.
C-----------------------------------------------------------------------
      [illegible]
      [illegible]
      [illegible]
      [illegible]
      [illegible]
[illegible] CONTINUE
[illegible] CONTINUE
C-----------------------------------------------------------------------
C     BEGINNING OF THE FAST COSINE TRANSFORM PROCESS TO FIND THE
C     WEIGHTING FUNCTION
C-----------------------------------------------------------------------
      [illegible]
      [illegible]
      [illegible]
      [illegible]
 60   CONTINUE
      CALL FFTSYM(X,N,[illegible])
      [illegible]
      [illegible]
 70   CONTINUE
 80   CONTINUE
      [illegible]
      [illegible]
      [illegible]
      [illegible]
[illegible] CONTINUE
      CALL FFTSYM([illegible])
      [illegible]
      [illegible]
[illegible] CONTINUE
[illegible] CONTINUE
C-----------------------------------------------------------------------
C     NOW THE FOURIER TRANSFORM OF THE CORRELATION FUNCTION IS
C     STORED IN ARRAY S(MX,MX). BY TAKING THE SQUARE ROOT AND
C     INVERSED FOURIER TRANSFORM WILL GIVE US THE WEIGHTING
C     FUNCTION.
C-----------------------------------------------------------------------
C
      [illegible]
```

```
      DO 90 J=1,NCOEF
        IF(S(I,J).LT.[illegible]) THEN
         S(I,J)=[illegible]
        ELSE
         S(I,J)=S(I,J)
        ENDIF
        S(I,J)=SQRT(S(I,J))
 90   CONTINUE
 100  CONTINUE


      DO 130 J=1,NCOEF
       DO 110 I=1,NCOEF
        X(I)=S(I,J)
        Y(I)=X
 110   CONTINUE
       CALL FT[illegible](X,N)
       DO 120 I=1,NCOEF
        S(I,J)=X(I)
 120   CONTINUE
 130  CONTINUE
      DO 131 I=1,NCOEF
       DO 111 J=1,NCOEF
        X(J)=S(I,J)
        Y(J)=X(J)
 111   CONTINUE
       CALL FT[illegible](X,N)
       DO 121 J=1,NCOEF
        [illegible]=X
 121   CONTINUE
 131  CONTINUE
C-----------------------------------------------------------------
C     NOW THE DESIRED WEIGHTING FUNCTION IS STORED IN ARRAY
C      S(NX,NY)
C     DUE TO THE OSCILLATING EFFECT AT LARGE LAG DISTANCE FROM
C     FFT OF THE FINITE LENGTH RECORD, ONLY THE DOMINANT PORTION
C     OF THE WEIGHTING FUNCTION IS TRANSFERRED FROM ARRAY S(NX,NY)
C     TO W(NWX,NWY)
C-----------------------------------------------------------------
      DO 140 I=1,NWYH
      W(NWXH,NWYH-I+1)=S(1,I)
      W(NWXH,NWYH+I-1)=S(1,I)
 140  CONTINUE
      DO 141 I=1,NWXH
      W(NWXH-I+1,NWYH)=S(I,1)
      W(NWXH+I-1,NWYH)=S(I,1)
 141  CONTINUE
      DO 150 I=2,NWXH
```

```
      DO 150 J=2,NWYH
       WK(NWXH+I-1,NWYH+J-1)=S(I,J)
       WK(NWXH-I+1,NWYH+J-1)=S(I,J)
       WK(NWXH-I+1,NWYH-J+1)=S(I,J)
       WK(NWXH+I-1,NWYH-J+1)=S(I,J)
  150 CONTINUE
  160 CONTINUE
C---------------------------------------------------------------------
C     GENERATE A STRIP OF RANDOM SURFACE ALONG X-DIRECTION
C---------------------------------------------------------------------
      NNX=NN*NX
      CALL ROUGH(WK,Z,NWX,NWY,NX,NY,NNX,NZX,NZY)
C---------------------------------------------------------------------
C     WRITE THE SURFACE TO DISK FILES IN PATCHES OF NN
C---------------------------------------------------------------------
      DO 180 J=1,NN
      JMIN=(J-1)*NX+1
      JMAX=JMIN+NX-1
      CALL OPENFILE(J)
       WRITE(2)((Z(K,L),L=1,NX),K=JMIN,JMAX)
  180 CONTINUE
      STOP
      END
C

      SUBROUTINE FFTSYM(X,N,Y)
      DIMENSION X(1),Y(1)
C
C    X=REAL ARRAY WHICH ON INPUT CONTAINS THE N/2+1 POINTS OF THE
C     INPUT SEQUENCE
C    Y=SCRATCH ARRAY OF SIZE N/2+2
C    FOR N=2,COMPUTES DFT DIRECTLY
C
C
      IF(N.GT.2) GO TO 10
      T=X(1)+X(2)
      X(2)=X(1)-X(2)
      X(1)=T
      RETURN
   10 TWOPI=8.*ATAN(1.)
C   FIRST COMPUTES B0 TERM,WHERE B0=SUM OF ODD VALUES OF X(I)
C
C
      NO2=N/2
      NO4=N/4
      NO2P=NO2+1
      B0=0.
      DO 20 I=2,NO2P,2
       B0=B0+X(I)
   20 CONTINUE
```

```
      B0=B0*2
C
      IF(N.EQ.4) GO TO 40
C     FORM NEW SEQUENCE,Y(M)=X(2*M)+(X(2*M+1)-X(2*M-1))
      DO 30 I=2,NO4
         IND=2*I
         T1=X(IND)-X(IND-2)
         Y(I)=X(IND-1)+T1
         IND1=NO2+2-I
         Y(IND1)=X(IND-1)-T1
   30 CONTINUE
   40 Y(1)=X(1)
      Y(NO4+1)=X(NO2+1)
C
C     TAKE N/2 POINTS (REAL) FFT OF Y
C
      CALL FAST(Y,NO2)
C
      TPN=TWOPI/FLOAT(N)
      COSI=2.*COS(TPN)
      SINI=2.*SIN(TPN)
      COSD=COSI/2
      SIND=SINI/2
      NIND=NO4+1
      DO 50 I=2,NIND
         IND=2*I
         BK=Y(IND)/SIN
         AK=Y(IND-1)
         X(I)=AK+BK
         NIND1=NO2+2-I
         X(NIND1)=AK-BK
         TEMP=COSI*COSD-SINI*SIND
         SINI=COSI*SIND+SINI*COSD
         COSI=TEMP
   50 CONTINUE
      X(1)=B0+Y(1)
      X(NO2+1)=Y(1)-B0
      RETURN
      END
C
C-----------------------------------------------------------------------C
C     SUBROUTINE IFTSYM
C-----------------------------------------------------------------------C
C

      SUBROUTINE IFTSYM(X,N,Y)
      DIMENSION X(1),Y(1)
```

```
C  FOR N=2,COMPUTES IDFT DIRECTLY
C
      IF(N.GT.2) GO TO 10
      T=(X(1)+X(2))/2
      X(2)=(X(1)-X(2))/2
      X(1)=T
      RETURN
  10  TWOPI=8.*ATAN(1.0)
C
C
      NO2=N/2
      NO4=N/4
      TPN=TWOPI/FLOAT(N)
      COSD=COS(TPN)
      SIND=SIN(TPN)
      COSI=2.0
      SINI=0.0
      X1=X(1)-X(NO2+1)
      DO 20 I=2,NO2
          TEMP=COSI*COSD-SINI*SIND
          SINI=COSI*SIND+SINI*COSD
          COSI=TEMP
          X1=X1+X(I)*COSI
  20  CONTINUE
      X1=X1/FLOAT(N)
C
C
      COSI=COS(TPN)
      SINI=SIN(TPN)
      COSD=COSI
      SIND=SINI
      X(1)=(X(1)+X(NO2+1))/2
      X(2)=0.0
      NNO=NO4+1
      DO 30 I=2,NNO
          INC=2*I
          NINC=NO2+2-I
          AK=(X(I)+X(NINC))/2
          BK=(X(I)-X(NINC))
          X(INC-1)=AK
          X(INC)=BK*SINI
          TEMP=COSI*COSD-SINI*SIND
          SINI=COSI*SIND+SINI*COSD
          COSI=TEMP
  30  CONTINUE
C
C  TAKE N/2 POINTS IDFT OF Y
C
```

```
      CALL FSST(X,N02)
C
      X(1)=Y(1)
      X(2)=X1
      IF(N.EQ.4) GO TO 50
      DO 40 I=2,N04
         IND=2*I
         IND1=N02+2-I
         X(IND-1)=(Y(I)+Y(IND1))/2
         T1=(Y(I)-Y(IND1))/2
         X(IND)=T1+X(IND-2)
  40  CONTINUE
  50  X(N02+1)=Y(N04+1)
      RETURN
      END
C
C
C-----------------------------------------------------------C
C     SUBROUTINE FAST                                       C
C     REPLACES THE REAL VECTOR B(K), FOR K=1,2,...,N,       C
C     WITH ITS FINITE DFT                                   C
C-----------------------------------------------------------C
C

      SUBROUTINE FAST(B,N)
      DIMENSION B(2)
      COMMON /CONS/ PII,P7,P7TWO,C22,S22,PI2
C
      PII=4.*ATAN(1.)
      PI8=PII/8.
      P7=1./SQRT(2.)
      P7TWO=2.*P7
      C22=COS(PI8)
      S22=SIN(PI8)
      PI2=2.*PII
      DO 10 I=1,15
         M=I
         NT=2**I
         IF(N.EQ.NT) GO TO 20
  10  CONTINUE
      WRITE(*,99)
  99  FORMAT(5X,'N IS NOT A POWER OF 2 FOR FAST')
      STOP
  20  N4POW=M/2
C
C
      IF(M-N4POW*2) 40,40,30
  30  NN=2
```

```
      INT=N/NN
      CALL FR2TR(INT,B(1),B(INT+1))
      GO TO 50
   40 NN=1
C
C
   50 IF(N4POW.EQ.0) GO TO 70
      DO 60 IT=1,N4POW
         NN=NN*4
         INT=N/NN
      CALL FR4TR(INT,NN,B(1),B(INT+1),B(2*INT+1),
     + B(3*INT+1),B(1),B(INT+1),B(2*INT+1),B(3*INT+1))
   60 CONTINUE
C
C     PERFORM IN-PLACE REORDERING
   70 CALL FORD1(M,B)
      CALL FORD2(M,B)
      T=B(2)
      B(2)=0.0
      B(N+1)=T
      B(N+2)=0.0
      DO 80 IT=4,N,2
         B(IT)=-B(IT)
   80 CONTINUE
      RETURN
      END

C-----------------------------------------------------------------------C
C     SUBROUTINE FSST
C-----------------------------------------------------------------------C
C
      SUBROUTINE FSST(B,N)
      DIMENSION B(2)
      COMMON /CONST/ PII,P7,P7TWO,C22,S22,PI2
C
      PII=4.*ATAN(1.)
      PI8=PII/8.
      P7=1./SQRT(2.)
      P7TWO=2.*P7
      C22=COS(PI8)
      S22=SIN(PI8)
      PI2=2.*PII
      DO 10 I=1,15
         M=I
         NT=2**I
         IF(N.EQ.NT) GO TO 20
   10 CONTINUE
      WRITE(*,9999)
```

```
 99   FORMAT(5X,'N IS NOT A POWER OF 2 FOR FSST')
      STOP
 20   B(2)=B(N+1)
      DO 30 I=4,N,2
          B(I)=-B(I)
 30   CONTINUE
C
C SCALE THE INPUT BY N
C
      DO 40 I=1,N
          B(I)=B(I)/FLOAT(N)
 40   CONTINUE
      N4POW=M/2
C
C  SCRAMBLE THE INPUTS
C
      CALL FORD2(M,B)
      CALL FORD1(M,B)
C
      IF(N4POW.EQ.0) GO TO 60
      NN=4*N
      DO 50 IT=1,N4POW
          NN=NN/4
          INT=N/NN
       CALL FR4SYN(INT,NN,B(1),B(INT+1),B(2*INT+1),B(3*INT+1),
     +            B(1),B(INT+1),B(2*INT+1),B(3*INT+1))
 50   CONTINUE
C
C DO A RADIX 2 ITERATION IF ONE IS REQUIRED
C
 60   IF(M-N4POW*2) 80,80,70
 70     INT=N/2
      CALL FR2TR(INT,B(1),B(INT+1))
 80   RETURN
      END
C
C
C----------------------------------------------------------------C
C     SUBROUTINE FR2TR                                    C
C     RADIX 2 ITERATION SUBROUTINE                              C
C----------------------------------------------------------------C
C
C
      SUBROUTINE FR2TR(INT,B0,B1)
      DIMENSION B0(2),B1(2)
      DO 10 K=1,INT
          T=B0(K)+B1(K)
          B1(K)=B0(K)-B1(K)
          B0(K)=T
 10   CONTINUE
```

```
      RETURN
      END
C
C-----------------------------------------------------------------------C
C     SUBROUTINE FR4TR                                  C
C     RADIX 4 ITERATION SUBROUTINE                          C
C-----------------------------------------------------------------------C
C
      SUBROUTINE FR4TR(INT,NN,B0,B1,B2,B3,B4,B5,B6,B7)
       DIMENSION L(15),B0(2),B1(2),B2(2),B3(2),B4(2),B5(2),B6(2),B7(2)
      COMMON /CONS/ PII,P7,P7TWO,C22,S22,PI2
       EQUIVALENCE (L15,L(1)),(L14,L(2)),(L13,L(3)),(L12,L(4)),
     +           (L11,L(5)),(L10,L(6)),(L9,L(7)),(L8,L(8)),
     +           (L7,L(9)),(L6,L(10)),(L5,L(11)),(L4,L(12)),
     +           (L3,L(13)),(L2,L(14)),(L1,L(15))
C
C
      L(1)=NN/4
      DO 40 K=2,15
         IF(L(K-1)-2) 10,20,30
 10       L(K-1)=2
 20       L(K)=2
         GO TO 40
 30       L(K)=L(K-1)/2
 40   CONTINUE
C
      PIOVN=PII/FLOAT(NN)
      JI=3
      JL=2
      JR=2
C
      DO 120 J1=2,L1,2
      DO 120 J2=J1,L2,L1
      DO 120 J3=J2,L3,L2
      DO 120 J4=J3,L4,L3
      DO 120 J5=J4,L5,L4
      DO 120 J6=J5,L6,L5
      DO 120 J7=J6,L7,L6
      DO 120 J8=J7,L8,L7
      DO 120 J9=J8,L9,L8
      DO 120 J10=J9,L10,L9
      DO 120 J11=J10,L11,L10
      DO 120 J12=J11,L12,L11
      DO 120 J13=J12,L13,L12
      DO 120 J14=J13,L14,L13
      DO 120 JTHET=J14,L15,L14
         TH2=JTHET-2
         IF(TH2) 50,50,90
```

```
50      DO 60 K=1,INT
        T0=B0(K)+B2(K)
        T1=B1(K)+B3(K)
        B2(K)=B0(K)-B2(K)
        B3(K)=B1(K)-B3(K)
        B0(K)=T0+T1
        B1(K)=T0-T1
60      CONTINUE
C
C
      IF(NN-4) 120,120,70
 70   K0=INT*4+1
      KL=K0+INT-1
      DO 80 K=K0,KL
         PR=P7*(B1(K)-B3(K))
         PI=P7*(B1(K)+B3(K))
         B3(K)=B2(K)+PI
         B1(K)=PI-B2(K)
         B2(K)=B0(K)-PR
         B0(K)=B0(K)+PR
 80   CONTINUE
      GO TO 120
C
C
 90   ARG=TH2*PIOVN
      C1=COS(ARG)
      S1=SIN(ARG)
      C2=C1**2-S1**2
      S2=C1*S1+C1*S1
      C3=C1*C2-S1*S2
      S3=C2*S1+S2*C1
C
      INT4=INT*4
      J0=JR*INT4+1
      K0=JI*INT4+1
      JLAST=J0+INT-1
      DO 100 J=J0,JLAST
         K=K0+J-J0
         R1=B1(J)*C1-B5(K)*S1
         R5=B1(J)*S1+B5(K)*C1
         T2=B2(J)*C2-B6(K)*S2
         T6=B2(J)*S2+B6(K)*C2
         T3=B3(J)*C3-B7(K)*S3
         T7=B3(J)*S3+B7(K)*C3
         T0=B0(J)+T2
         T4=B4(K)+T6
         T2=B0(J)-T2
         T6=B4(K)-T6
```

```
        T1=R1+T3
        T5=R5+T7
        T3=R1-T3
        T7=R5-T7
        B0(J)=T0+T1
        B7(K)=T4+T5
        B6(K)=T0-T1
        B1(J)=T5-T4
        B2(J)=T2-T7
        B5(K)=T6+T3
        B4(K)=T2+T7
        B3(J)=T3-T6
 100  CONTINUE
C
      JR=JR+2
      JI=JI-2
      IF(JI-JL) 110,110,120
 110  JI=2*JR-1
      JL=JR
 120  CONTINUE
      RETURN
      END
C
C-----------------------------------------------------------C
C   SUBROUTINE FR4SYN                    C
C   RADIX 4 SYNTHESIS                    C
C-----------------------------------------------------------C
C
C
C
      SUBROUTINE FR4SYN(NTHPO,NN,B0,B1,B2,B3,B4,B5,B6,B7)
       DIMENSION L(15),B0(2),B1(2),B2(2),B3(2),B4(2),B5(2)
      DIMENSION B6(2),B7(2)
      COMMON /CONST/ PII,P7,P7TWO,C22,S22,PI2
       EQUIVALENCE (L15,L(1)),(L14,L(2)),(L13,L(3)),(L12,L(4)),
     +          (L11,L(5)),(L10,L(6)),(L9,L(7)),(L8,L(8)),
     +          (L7,L(9)),(L6,L(10)),(L5,L(11)),(L4,L(12)),
     +          (L3,L(13)),(L2,L(14)),(L1,L(15))
C
      L(1)=NN/4
      DO 40 K=2,15
         IF(L(K-1)-2) 10,20,30
  10     L(K-1)=2
  20     L(K)=2
         GO TO 40
  30     L(K)=L(K-1)/2
  40  CONTINUE
C
      PIOVN=PII/FLOAT(NN)
```

```
      JI=3
      JL=2
      JR=2
C
      DO 120 J1=2,L1,2
      DO 120 J2=J1,L2,L1
      DO 120 J3=J2,L3,L2
      DO 120 J4=J3,L4,L3
      DO 120 J5=J4,L5,L4
      DO 120 J6=J5,L6,L5
      DO 120 J7=J6,L7,L6
      DO 120 J8=J7,L8,L7
      DO 120 J9=J8,L9,L8
      DO 120 J10=J9,L10,L9
      DO 120 J11=J10,L11,L10
      DO 120 J12=J11,L12,L11
      DO 120 J13=J12,L13,L12
      DO 120 J14=J13,L14,L13
      DO 120 JTHET=J14,L15,L14
         TH2=JTHET-2
         IF(TH2) 50,50,90
 50   DO 60 K=1,INT
         T0=B0(K)+B1(K)
         T1=B0(K)-B1(K)
         T2=B2(K)*2.0
         T3=B3(K)*2.0
         B0(K)=T0+T2
         B2(K)=T0-T2
         B1(K)=T1+T3
         B3(K)=T1-T3
 60   CONTINUE
C
      IF(NN-4) 120,120,70
 70   K0=INT*4+1
      KL=K0+INT-1
      DO 80 K=K0,KL
         T2=B0(K)-B2(K)
         T3=B1(K)+B3(K)
         B0(K)=(B0(K)+B2(K))*2.0
         B2(K)=(B3(K)-B1(K))*2.0
         B1(K)=(T2+T3)*P7TWO
         B3(K)=(T3-T2)*P7TWO
 80   CONTINUE
      GO TO 120
 90   ARG=TH2*PIOVN
      C1=COS(ARG)
      S1=-SIN(ARG)
      C2=C1**2-S1**2
```

```
      S2=C1*S1+C1*S1
      C3=C1*C2-S1*S2
      S3=C2*S1+S2*C1
C
      INT4=INT*4
      J0=JR*INT4+1
      K0=JI*INT4+1
      JLAST=J0+INT-1
      DO 100 J=J0,JLAST
         K=K0+J-J0
         T0=B0(J)+B6(K)
         T1=B7(K)-B1(J)
         T2=B0(J)-B6(K)
         T3=B7(K)+B1(J)
         T4=B2(J)+B4(K)
         T5=B5(K)-B3(J)
         T6=B5(K)+B3(J)
         T7=B4(K)-B2(J)
         B0(J)=T0+T4
         B4(K)=T1+T5
         B1(J)=(T2+T6)*C1-(T3+T7)*S1
         B5(K)=(T2+T6)*S1+(T3+T7)*C1
         B2(J)=(T0-T4)*C2-(T1-T5)*S2
         B6(K)=(T0-T4)*S2+(T1-T5)*C2
         B3(J)=(T2-T6)*C3-(T3-T7)*S3
         B7(K)=(T2-T6)*S3+(T3-T7)*C3
100   CONTINUE
      JR=JR+2
      JI=JI-2
      IF(JI-JL) 110,110,120
110   JI=2*JR-1
      JL=JR
120   CONTINUE
      RETURN
      END
C
C---------------------------------------------------------------C
C     SUBROUTINE FORD1                                          C
C     IN-PLACE REORDERING SUBROUTINE                            C
C---------------------------------------------------------------C
C
      SUBROUTINE FORD1(M,B)
      DIMENSION B(2)
C
      K=4
      KL=2
      N=2**M
      DO 40 J=4,N,2
```

```
      IF(K-J)20,20,10
10    T=B(J)
      B(J)=B(K)
      B(K)=T
20    K=K-2
      IF(K-KL) 30,30,40
30    K=2*J
      KL=J
40    CONTINUE
      RETURN
      END
C
C-----------------------------------------------------------C
C     SUBROUTINE FORD2                          C
C     IN-PLACE REORDERING SUBROUTINE                   C
C-----------------------------------------------------------C
C
      SUBROUTINE FORD2(M,B)
      DIMENSION L(15),B(2)
      EQUIVALENCE (L15,L(1)),(L14,L(2)),(L13,L(3)),(L12,L(4)),
     +     (L11,L(5)),(L10,L(6)),(L9,L(7)),(L8,L(8)),
     +     (L7,L(9)),(L6,L(10)),(L5,L(11)),(L4,L(12)),
     +     (L3,L(13)),(L2,L(14)),(L1,L(15))
      N=2**M
      L(1)=N
      DO 10 K=2,M
        L(K)=L(K-1)/2
10    CONTINUE
      DO 20 K=M,14
        L(K+1)=2
20    CONTINUE
      IJ=2
      DO 40 J1=2,L1,2
      DO 40 J2=J1,L2,L1
      DO 40 J3=J2,L3,L2
      DO 40 J4=J3,L4,L3
      DO 40 J5=J4,L5,L4
      DO 40 J6=J5,L6,L5
      DO 40 J7=J6,L7,L6
      DO 40 J8=J7,L8,L7
      DO 40 J9=J8,L9,L8
      DO 40 J10=J9,L10,L9
      DO 40 J11=J10,L11,L10
      DO 40 J12=J11,L12,L11
      DO 40 J13=J12,L13,L12
      DO 40 J14=J13,L14,L13
      DO 40 JI=J14,L15,L14
        IF(IJ-JI) 30,40,40
```

DATE
ILMED
8